PowerModelsDistribution.jl: An Open-Source Framework for Exploring Distribution Power Flow Formulations

David M Fobes*, Sander Claeys†, Frederik Geth‡, and Carleton Coffrin*

* Los Alamos National Laboratory (LANL)
Los Alamos, New Mexico, USA
{dfobes,cjc}@lanl.gov

† Katholieke Universiteit Leuven (KU Leuven)
Leuven, Belgium
sander.claeys@kuleuven.be
,

‡ Commonwealth Scientific and Industrial Research Organisation (CSIRO)
Canberra, Australia
frederik.geth@csiro.au

Abstract—In this work we introduce PowerModelsDistribution, a free, open-source toolkit for distribution power network optimization, whose primary focus is establishing a baseline implementation of steady-state multi-conductor unbalanced distribution network optimization problems, which includes implementations of Power Flow and Optimal Power Flow problem types. Currently implemented power flow formulations for these problem types include AC (polar and rectangular), a second-order conic relaxation of the Branch Flow Model (BFM) and Bus Injection Model (BIM), a semi-definite relaxation of BFM, and several linear approximations, such as the simplified unbalanced BFM. The results of AC power flow have been validated against OpenDSS, an open-source "electric power distribution system simulator", using IEEE distribution test feeders (13, 34, 123 bus and LVTestCase), all parsed using a built-in OpenDSS parser. This includes support for standard distribution system components as well as novel resource models such as generic energy storage (multi-period) and photovoltaic systems, with the intention to add support for additional components in the future.

Index Terms—nonlinear optimization, convex optimization, AC optimal power flow, Julia Language, Open-Source

I. INTRODUCTION

A. Background

The past few decades has seen a remarkable increase in the penetration of distributed energy resources (DER), such as photovoltaic systems, wind turbines, storage devices, and fuel cells, as well as the unique demands of controllable loads, such as electric vehicle chargers and heating, ventilation and air conditioning (HVAC) systems, within electric distribution networks. While the application of DERs can mitigate the need for more traditional transmission network expansion, *i.e.*, via the addition of generators, the operation and control of distribution networks remains a challenge. In recent years this has driven considerable interest in the research community to develop multi-conductor power network optimization formulations for various applications, such as a networked micro-grid design. While the number of mathematical formulations for distribution system modeling has increased, few open-source tools are yet available, and none that enable rapid development of the newest formulations and optimization problems, to the best of our knowledge.

B. The Development of PowerModels

Recently, in response to an explosion of the number of power flow approximations and relaxations appearing in the literature for transmission networks, PowerModels [1] was offered as a free, open-source toolkit for the optimization of steady-state power transmission networks. Written in Julia, a high-level high-performance programming language for numerical computing, and utilizing JuMP [2], PowerModels provides a powerful expansive modeling layer for optimization; PowerModels is engineered to decouple problem specifications, *e.g.*, Optimal Power Flow (OPF) or Optimal Transmission Switching (OTS), from formulations, *e.g.*, AC or second-order cone (SOC) relaxations. This decoupled design allows for more accurate, faster, and more detailed comparisons between the variety of problem specifications and formulations that are constantly emerging from the research community.

PowerModels, being easily extensible, has generated a number of spin-off packages, for modeling, *e.g.*, DC networks and HVDC converters [3], maximum load delivery under contingencies [4], or the impact of geomagnetic disturbances[1].

C. Introducing PowerModelsDistribution

Building on the success of PowerModels in the area of *transmission* power network optimization, in this work we introduce PowerModelsDistribution, a free, open-source toolkit for *distribution* power network optimization developed on top of JuMP, a mathematical programming abstraction layer for

---

[1]https://github.com/lanl-ansi/PowerModelsGMD.jl

optimization. This project currently focuses on establishing a baseline implementation of steady-state multi-conductor unbalanced distribution network optimization problems. Many recently published formulations (for an overview, see [5]) are already included, from non-convex nonlinear forms, to convex relaxations, to linear approximations of the unbalanced power flow equations. Although both approximations and relaxations might not satisfy the originating formulation, e.g. non-linear power flow equations, unlike approximations, relaxations can provide information about the original problem. For example, when a relaxation is infeasible, it certifies that the original problem is also infeasible. Furthermore, relaxations can also provides a lower bound on the objective, which can be useful for global optimization methods [5]. Recent work extends relaxations of the power flow equations to multi-phase networks [6], [7], [8].

PowerModelsDistribution is a tool designed to build and compare unbalanced power network *optimization* problems, *e.g.*, Optimal Power Flow. In this work, to demonstrate the accuracy of our mathematical formulations, we focus on the results of power flow feasibility; the numerical results of the AC power flow are validated w.r.t. OpenDSS on IEEE distribution test cases [9], which include a broad range of distribution network components. While we support the OpenDSS data format as input, it is not the intention of this package to replicate the features of OpenDSS, but to leverage their existing mature data format for building network cases. Within its robust OpenDSS parser, PowerModelsDistribution supports a subset of DER, including generic energy storage and photovoltaic systems, with support for additional DER components coming in the future. In time, we hope that PowerModelsDistribution will emerge as an essential part of the distribution network *optimization* and analysis toolkit.

This paper therefore explores the design of an open-source toolkit for simulation and optimization of distribution network power flows, where careful consideration of phase unbalance effects is important. Modeling focuses on (quasi-)steady state physics, described in the frequency domain.

II. REVIEW OF DISTRIBUTION MODELING APPROACHES

For a recent, in-depth discussion of free, open-source power system simulation and optimization tools we refer to [10]; here we focus on the subset of tools that are capable of representing the effects of unbalanced phases. Phase unbalance is the consequence of unbalanced loading, such as single-phase loads or generators connected to a three-phase network, or unequal branch impedance, *e.g.*, caused by a lack of transposition of conductors with a non-equilateral geometry.

Calculation of the power flow in networks featuring non-negligible phase unbalance has long been a topic of interest [11]. Several open-source tools include algorithms for solving unbalanced power flows: OpenDSS [12], originally developed at the Electrotek Concepts in 1997, and open-sourced at the Electric Power Research Institute in 2008, was one of the first free tools for distribution system simulation; in 2003, the Pacific Northwest National Laboratory started development of

Gridlab-D [13]; recently, PandaPower [10] and GridCal [14] have added support for unbalanced power flow.

A variety of commercial options is also available and in use globally by distribution system operators for solving power flows, including PowerFactory, PSS/Sincal, Neplan, and CYME. While most of these rely on proprietary file formats, several conversion tools have been developed to enable data exchange:
- DiTTo[2], supports conversion between a large variety of open-source and commercial tools;
- GRIDAPPSD/Powergrid-Models[3] provides CIM interfaces for GridLAB-D and OpenDSS.

Fortunately, standardization of data models for distribution system management (CIM Common Distribution Power System Model) is being developed as part of IEC 61968 [15].

A. Foundations of Distribution Network Modeling

The physics of power flow in networks with phase unbalance are described in the framework of Kirchhoff's circuit laws; the equations are linear in current and voltage for fixed impedances. In the context of modeling the coupling between conductors, the impedance becomes a matrix parameter composed of self-impedance (diagonal) and mutual impedances (off-diagonals). Therefore, the current-voltage variable space represents a natural choice for distribution modeling.

The size of the impedance matrix varies on the context; three-wire and Kron-reduced four-wire networks use 3×3 impedance matrices, but generalizations that include an explicit neutral conductor voltage and current variables, or even earth voltage rise effects [16], can result in up to 5×5 impedance matrices. In practice, in the collection of data and the construction of the mathematical models a variety of approximations are made [17], including:
- neglecting branch shunt admittance,
- assuming constant-power load behavior,
- assuming perfect grounding of the neutral at all buses,
- assuming loads are balanced over the phases.

To support nontrivial edge cases, PowerModelsDistribution has chosen generic branch and bus representations; formulations are standardized on multi-conductor Π sections, supporting full matrices for the series and shunt elements (variable size). Furthermore, the admittance shunts on either side of the conductor are not necessarily identical, thereby enabling Γ sections. The branch model of PowerModelsDistribution is illustrated in Fig. 1.

In the following, we illustrate the primary features of the physics of unbalanced power flow in the current-voltage variable space, assuming constant impedances.

1) Voltage and Current Variables: The bus voltage vector $\mathbf{U}_i$ stacks the scalar complex-value voltages for each phase,

$$\mathbf{U}_i = \begin{bmatrix} U_{i,a} \\ U_{i,b} \\ U_{i,c} \end{bmatrix} = \begin{bmatrix} U_{i,a}^{\text{re}} \\ U_{i,b}^{\text{re}} \\ U_{i,c}^{\text{re}} \end{bmatrix} + j \begin{bmatrix} U_{i,a}^{\text{im}} \\ U_{i,b}^{\text{im}} \\ U_{i,c}^{\text{im}} \end{bmatrix}. \qquad (1)$$

[2]https://github.com/NREL/ditto
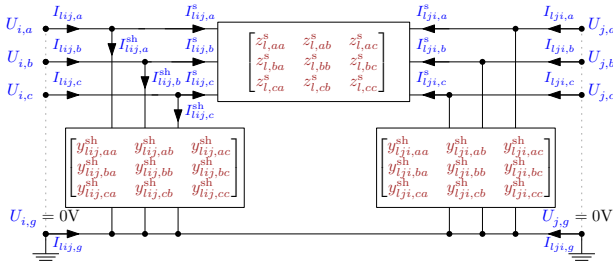[3]https://github.com/GRIDAPPSD/Powergrid-Models

Fig. 1. Unbalanced 3×3 Π-model branch in voltage and current variables.

Similarly, series and shunt current vectors are defined:

$$\mathbf{I}_{lij} = \begin{bmatrix} I_{lij,a} \\ I_{lij,b} \\ I_{lij,c} \end{bmatrix} = \begin{bmatrix} I^{\mathrm{s}}_{lij,a} \\ I^{\mathrm{s}}_{lij,b} \\ I^{\mathrm{s}}_{lij,c} \end{bmatrix} + \begin{bmatrix} I^{\mathrm{sh}}_{lij,a} \\ I^{\mathrm{sh}}_{lij,b} \\ I^{\mathrm{sh}}_{lij,c} \end{bmatrix} = \mathbf{I}^{\mathrm{s}}_{lij} + \mathbf{I}^{\mathrm{sh}}_{lij}. \quad (2)$$

2) Ohm's Law: The voltage at node j w.r.t. node i, and the drop over the series impedance $\mathbf{Z}_l$ in a branch l is,

$$\mathbf{U}_j = \mathbf{U}_i - \mathbf{Z}_l \mathbf{I}^{\mathrm{s}}_{lij}. \quad (3)$$

The branch shunt admittance $\mathbf{Y}^{\mathrm{sh}}_{lij}$ draws a current,

$$\mathbf{I}^{\mathrm{sh}}_{lij} = \mathbf{Y}^{\mathrm{sh}}_{lij} \mathbf{U}_i. \quad (4)$$

3) Kirchhoff's Current Law: We can substitute (3) and (4) into (2) to obtain the total current as a function of voltage, with $\mathbf{Y}^{\mathrm{s}}_l = (\mathbf{Z}^{\mathrm{s}}_l)^{-1}$:

$$\mathbf{I}_{lij} = \mathbf{Y}^{\mathrm{sh}}_{lij} \mathbf{U}_i + \mathbf{Y}^{\mathrm{s}}_l (\mathbf{U}_i - \mathbf{U}_j). \quad (5)$$

At buses, current flows between the branches $lij \in \mathcal{T}$, bus shunts $b \in \mathcal{B}$, loads $d \in \mathcal{D}$ and generators $g \in \mathcal{G}$ connected,

$$\sum_{lij \in \mathcal{T}(i)} \mathbf{I}_{lij} = \sum_{b \in \mathcal{B}(i)} \mathbf{I}^{\mathrm{sh}}_b + \sum_{g \in \mathcal{G}(i)} \mathbf{I}_g - \sum_{d \in \mathcal{D}(i)} \mathbf{I}_d. \quad (6)$$

4) Complex Power Flow and Sources/Sinks: The power flow into a branch is defined

$$\mathbf{S}_{lij} = \mathbf{U}_i (\mathbf{I}_{lij})^{\mathrm{H}}, \quad (7)$$

Note that this equation is nonlinear and non-convex, *i.e.*, the product of the voltage vector at the bus is multiplied by the conjugate transpose of the current vector. This results in $\mathbf{S}_{lij}$ being a matrix, whose diagonal contains the elements $S_{lij,pp} = U_{i,p}(I_{lij,p})^* \ \forall p \in \{a, b, c\}$. Analogously, we define the power of loads and generators,

$$\mathbf{S}_d = \mathbf{U}_i (\mathbf{I}_d)^{\mathrm{H}} \quad (8)$$
$$\mathbf{S}_g = \mathbf{U}_i (\mathbf{I}_g)^{\mathrm{H}}. \quad (9)$$

If generators are dispatchable, we define bounds on $diag(\mathbf{S}_g)$.

5) Matrix, Vector and Scalar Forms: Next, we substitute (5) in into (7), obtaining the *matrix* equality,

$$\mathbf{S}_{lij} = \mathbf{U}_i (\mathbf{U}_i)^{\mathrm{H}} (\mathbf{Y}^{\mathrm{sh}}_{lij})^{\mathrm{H}} + \mathbf{U}_i (\mathbf{U}_i - \mathbf{U}_j)^{\mathrm{H}} (\mathbf{Y}^{\mathrm{s}}_{lij})^{\mathrm{H}}. \quad (10)$$

Although in general the off-diagonal terms in (10) are redundant (*i.e.*, rotated/scaled versions of the diagonal terms), when performing lifting and convex relaxation, these off-diagonal terms are not necessarily redundant. Dropping the off-diagonal terms results in a *vector* form. Finally, the diagonal terms are *scalarized* and converted to reals before implementation, *e.g.*, for the active power flow, parameterized on the diagonal elements p by:

$$P_{lij,pp} = \sum_{q \in \mathcal{P}} (U^{\mathrm{re}}_{i,p} U^{\mathrm{re}}_{i,q} + U^{\mathrm{im}}_{i,p} U^{\mathrm{im}}_{i,q}) (g^{\mathrm{s}}_{l,pq} + g^{\mathrm{sh}}_{lij,pq})$$
$$+ \sum_{q \in \mathcal{P}} (U^{\mathrm{im}}_{i,p} U^{\mathrm{re}}_{i,q} - U^{\mathrm{re}}_{i,p} U^{\mathrm{im}}_{i,q}) (b^{\mathrm{s}}_{l,pq} + b^{\mathrm{sh}}_{lij,pq})$$
$$- \sum_{q \in \mathcal{P}} (U^{\mathrm{re}}_{i,p} U^{\mathrm{re}}_{j,q} + U^{\mathrm{im}}_{i,p} U^{\mathrm{im}}_{j,q}) g^{\mathrm{s}}_{l,pq}$$
$$- \sum_{q \in \mathcal{P}} (U^{\mathrm{im}}_{i,p} U^{\mathrm{re}}_{j,q} + U^{\mathrm{re}}_{i,p} U^{\mathrm{im}}_{j,q}) b^{\mathrm{s}}_{l,pq}. \quad (11)$$

Note that alternatively, this equation could be derived using polar voltage variables instead. These alternatives can have different numerical properties, and different relaxations can be obtained when starting from different forms.

B. Separating Modeling and Solving

Now understanding the physics-based foundations, we ask whether distribution system models can be stated as mathematical optimization programs, referred to henceforth simply as mathematical programs. Mathematical modeling tools allow for a clear separation of the model and the solution method, which results in a declarative programming approach, as the modeler merely describes the problem structure, and requests the solver to return a solution satisfying a set of conditions without prescribing *how* the solution should be found. Well-known mathematical modeling layers include: GAMS, AMPL, AIMMS, Yalmip, JuMP, Pyomo, CVXPy, and Convex. These modeling layers access the different solution algorithms, *e.g.*, solvers for nonlinear programming (NLP), semidefinite programming (SDP), second-order cone programming (SOCP), or linear programming (LP) problems, which allows for the ability to switch solution approaches quickly without having to re-implement the equations. This is an essential feature for the rapid prototyping that is central to algorithmic research, allowing for direct comparison of solver performance on identical problems and formulations.

Such an approach has not been commonly used in distribution system modeling; the bulk of the work to date on distribution modeling has focused on power flow *algorithms* describing processes for computing solutions to specific problems (*e.g.*, unbalanced power flow), as opposed to optimization problems on which we focus on here. When it comes to solving power flow problems, it is an open question whether such mathematical programs for optimization can reliably produce

results with similar accuracy and modeling fidelity as state-of-the-art distribution power flow tools such as GridLab-D and OpenDSS. Due to this approach, the performance of the power flow solve is much less than that achieved by dedicated power flow solvers by design, and is not meant as a replacement for dedicated power flow, but to serve as a means of fidelity comparison between other tools. In PowerModelsDistribution, the power flow is obtained via the same interface as for example OPF, utilizing constrained optimization solvers.

To facilitate collaboration, mathematical modeling layers should enable swift implementation and maintainability. Features that are important in this context are:

- supporting vector / matrix equations (automatic scalarization),
- automatic derivative computations of nonlinear equations, and
- support for semidefinite variables and/or constraints,

all of which are supported by the Julia-based mathematical programming package JuMP [2].

III. DISTRIBUTION SYSTEM MATHEMATICAL PROGRAMS

The following section provides an overview of the components and modeling assumptions of PowerModelsDistribution and describes several approaches to the mathematical formulation of distribution network physics.

A. Component Models

In distribution networks, a broad variety of power transfer and consumption/generation sources are encountered. This section details which components are prioritized, and which mathematical modeling aspects are considered.

1) Buses and nodes (`bus`): Buses have multiple nodes, *i.e.*, one for each conductor. In optimization we want to optimally operate the network such that voltage limits are satisfied. The voltage magnitudes are thus given lower and upper bounds,

$$\begin{bmatrix} U_{i,a}^{\min} \\ U_{i,b}^{\min} \\ U_{i,c}^{\min} \end{bmatrix} \leq \begin{bmatrix} |U_{i,a}| \\ |U_{i,b}| \\ |U_{i,c}| \end{bmatrix} \leq \begin{bmatrix} U_{i,a}^{\max} \\ U_{i,b}^{\max} \\ U_{i,c}^{\max} \end{bmatrix}, \quad (12)$$

typically chosen in accordance with grid codes. Note that bounds can also be applied to the sequence components. For instance, the ratio of the negative to positive sequence components of the voltage phasor should be below 2% [18].

2) Cables and overhead lines (`branch`): Both cables and overhead lines are represented through multi-conductor Π sections (Fig. 1). Flow bounds, on current or power magnitude, are applied to both ends of a Π-section,

$$\begin{bmatrix} |I_{lij,a}| \\ |I_{lij,b}| \\ |I_{lij,c}| \end{bmatrix} \leq \begin{bmatrix} I_{lij,a}^{\max} \\ I_{lij,b}^{\max} \\ I_{lij,c}^{\max} \end{bmatrix}, \begin{bmatrix} |S_{lij,aa}| \\ |S_{lij,bb}| \\ |S_{lij,cc}| \end{bmatrix} \leq \begin{bmatrix} S_{lij,aa}^{\max} \\ S_{lij,bb}^{\max} \\ S_{lij,cc}^{\max} \end{bmatrix}. \quad (13)$$

The Π sections are easily generalized for a different number of conductors. Reductions in number of conductors (*e.g.*, a single-phase branch off the feeder) are also supported.

3) Bus shunts (`shunt`): Bus shunts $b \in \mathcal{B}$, defined through an admittance $\mathbf{Y}_b^{\text{sh}}$, can be used to represent components such as capacitor banks,

$$\mathbf{I}_b^{\text{sh}} = \mathbf{Y}_b^{\text{sh}} \mathbf{U}_i. \quad (14)$$

4) Transformers (`transformer`): Transformers, which are different from voltage regulators due to their galvanic isolation, can have delta, wye and zigzag winding configurations, with two or more winding sets. The choice of configuration leads to a phase angle off-set between the primary and secondary (*i.e.*, vector group), in multiples of 30°. If so equipped, the neutral can be grounded. No-load losses can be significant, and are therefore modeled explicitly. PowerModelsDistribution implements the comprehensive approach to transformer modeling detailed in [19], where n-winding transformers are decomposed into a set of simple idealized two-winding transformers and lossy branches.

A simple idealized transformer, index t, modeled with transformation matrix $\mathbf{T}_t$, is

$$\mathbf{U}_i = \mathbf{T}_t \mathbf{U}_j, \quad \mathbf{T}_t^{\text{H}} \mathbf{I}_{tij} + \mathbf{I}_{tji} = 0. \quad (15)$$

Transformers can also be equipped with on-load tap changers, which can be either operated jointly (gang operation), or separately for each phase.

5) Loads, Generators, DER and Storage (`load`, `gen`, `storage`): Power consumption and generation devices can be configured with wye or delta connections; phase-to-neutral configurations are special cases of wye connections.

In practice, instantaneous load power depends on the voltage magnitude; such voltage-dependent behavior is often described through ZIP or exponential load models.

While photovoltaic components are currently approximated as fixed generators in the internal data model where the maximum real and reactive power generation capability can vary over time, storage systems can be modeled using PowerModels' *multi-network* feature, which enables multi-period optimization models; storage systems are modeled generically in such a way as to be able to represent a variety of different types of storage, like batteries, fuel cells, or flywheels [4].

6) Switches (`switch`): Breakers, fuses, switches or sectionalizers allow for altering the topology of an existing network. Switches can be fixed open, fixed closed, or be optimized.

B. Unbalanced OPF Formulations

The unbalanced formulations currently implemented in PowerModelsDistribution are detailed in Table I, including their category (cat.), their variable space (var.), their coordinate space (coord.), their mathematical complexity (compl.), and their functional representation (repr.). These classifications are discussed in detail below. Note that each formulation collects mathematical formulations for all the components discussed, not solely for branches.

The upcoming subsections detail why and how these formulations are categorized.

---

[4]https://lanl-ansi.github.io/PowerModels.jl/stable/storage

TABLE I
FEATURES OF UNBALANCED POWER FLOW FORMULATIONS

Name	cat.	var.	coord.	compl.	repr.	ref.
ACPPowerModel	BIM	SU	polar	NLP	trig.	[20]
ACRPowerModel	BIM	SU	rect.	NLP	quadr.	
IVRPowerModel	BIM	IU	rect.	NLP	quadr.	
SDPUBFPowerModel	BFM	SW	rect.	SDP	conic	[21]
SOCConicUBFPowerModel	BFM	SW	rect.	SOC	conic	[21], [22]
SOCNLPUBFPowerModel	BFM	SW	rect.	SOC	quadr.	[21], [22]
LPUBFDiagPowerModel	BFM	SW	rect.	LP	lin.	[21], [23], [24]
DCPPowerModel	BIM	SU	polar	LP	lin.	

1) Category: We can categorize formulations into either a bus injection model (BIM) or a branch flow model (BFM) [25].

2) Variable Space: While the component models were illustrated in current-voltage vector variables, the same physics can be represented in different variable spaces. This mechanism is used to obtain mathematical equations which have different behavior when solved with numerical methods. We categorize some recent work in mathematical formulations for unbalanced OPF depending on the choice of variable space for the branch flow equations into three distinct sets:

- current $\mathbf{I}_{lij}$ - voltage $\mathbf{U}_i$,
- power $\mathbf{S}_{lij}$ - voltage $\mathbf{U}_i$,
- power $\mathbf{S}_{lij}$ - lifted voltage $\mathbf{W}_{ij} = \mathbf{U}_i\mathbf{U}_j$.

3) Coordinates: Either rectangular or polar coordinates can represent complex voltage variables.

4) Mathematical Complexity: We can have continuous convex and non-convex formulations of different complexities,

$$\text{NLP} \supset \text{SDP} \supset \text{SOC} \supset \text{LP}. \tag{16}$$

5) Representation: We can distinguish quadratic and conic representations of convex forms. For example, given $x_3 \geq 0, x_4 \geq 0$:

$$(x_1)^2 + (x_2)^2 \leq x_3 x_4 \iff \left\| \begin{bmatrix} 2x_1 \\ 2x_2 \\ x_3 - x_4 \end{bmatrix} \right\| \leq x_3 + x_4.$$

Note that linear constraints are both quadratic and conic and therefore only have one representation. The polynomial form can be used in conjunction with gradient-based solvers, whereas SOC solvers have norm-based interfaces. Trigonometric expressions, such as the AC polar form power flow equations, which uses sine and cosine, are not polynomial.

IV. USING POWERMODELSDISTRIBUTION

Following the design of PowerModels [1], each power system optimization problem (*e.g.*, unbalanced OPF) has well-defined semantics for a large set of formulations (*e.g.*, AC in polar coordinates, DC approximation, or SOC relaxation).

A. Abstract Problem Definitions

Several key problem definitions are already included in PowerModelsDistribution. In particular, Power Flow, Optimal Power Flow, and Maximal Load Delivery [4]. Code Block 1 shows a typical example of the OPF problem definition; first,

Code Block 1 Problem specification for unbalanced OPF.

```
function post_mc_opf(pm::AbstractPowerModel)
    variable_mc_voltage(pm)
    variable_mc_branch_flow(pm)
    variable_mc_transformer_flow(pm)
    variable_mc_generation(pm)
    variable_mc_storage(pm)

    constraint_mc_model_voltage(pm)

    for i in ids(pm, :ref_buses)
        constraint_mc_theta_ref(pm, i)
    end

    for i in ids(pm, :bus)
        constraint_mc_power_balance(pm, i)
    end

    for i in ids(pm, :storage)
        constraint_storage_state(pm, i)
        constraint_storage_complementarity_nl(pm, i)
        constraint_mc_storage_loss(pm, i)
        constraint_mc_storage_thermal_limit(pm, i)
    end

    for i in ids(pm, :branch)
        constraint_mc_ohms_yt_from(pm, i)
        constraint_mc_ohms_yt_to(pm, i)

        constraint_mc_voltage_angle_difference(pm, i)

        constraint_mc_thermal_limit_from(pm, i)
        constraint_mc_thermal_limit_to(pm, i)
    end

    for i in ids(pm, :transformer)
        constraint_mc_trans(pm, i)
    end

    objective_min_fuel_cost(pm)
end
```

variables for voltage, branch and transformer flows, generators and storage are initialized for the specified formulation, and then constraints are applied, including constraints for power balance, Ohm's Law, thermal limits, etc. Finally, a objective function, in this case a standard minimum fuel cost objective, is added to the problem. Details of each JuMP model will vary according to chosen formulation.

B. Abstract Formulations

Starting from a generic specification of an unbalanced OPF problem, PowerModels demonstrated that it is possible to specialize the model into concrete mathematical programs for given power flow formulations [1]. Formulations define a representation of the electrical physics in a certain variable space (*e.g.*, power-voltage), with a choice of coordinates (*e.g.*, polar or rectangular). PowerModelsDistribution maintains the assumption that the combination of an abstract problem and a mathematical formulation results in a fully specified mathematical program, which is encoded as a JuMP model. Independently, the user can select a JuMP-compatible solver to solve the program.

C. Data Formats

During initial development we considered support for a variety of existing network data formats, ultimately settling on the OpenDSS format, which exhibits a good balance between its detail and inherent readability. In PowerModelsDistribution, only a subset of the OpenDSS format is supported; in particular, we support a subset of components for conversion into the internal data model, which follows closely that of PowerModels [1], including Loads, Capacitors, Reactors (only as shunts, not generically), Lines and Linecodes, Transformers, Generators, PVSystems, and Storage. It should also be noted that like PowerModels, and unlike OpenDSS, PowerModelsDistribution uses a non-dimensional unit system (per-unit) in its internal model to facilitate numerical stability.

In general, OpenDSS functions are not supported, with the notable exception of data-handling functions such as setting properties, and redirecting to additional files. That being said, the parser can ingest any valid OpenDSS files into a serializable data structure for additional user processing. Furthermore, several advanced input styles are fully supported, such as upper triangular matrices and reverse polish notation. Additional notes about support for the OpenDSS format can be found in the package documentation.[5]

V. PROOF-OF-CONCEPT STUDY

A. Test Cases and Computational Setting

This study considers 4 of the IEEE unbalanced test feeders, *i.e.*, IEEE 13, 34, 123 and LVTestCase [9]. We use the '.dss' case definitions, to facilitate comparison of power flow results w.r.t. OpenDSS. Comparing formulations is a multifaceted endeavor, requiring both high quality implementations of the mathematical models, as well as non-trivial test cases; the toolbox needs to facilitate comparisons of feasibility, optimality, computation time, memory usage and reliability (convergence).

In Table II we highlight the core features used in the IEEE distribution feeder test cases. All cases are Kron-reduced and do not feature explicit neutrals. The table shows the types of branch shunts (branch sh.), noting whether they are full matrices, diagonal matrices, or contain none, the types of transformers (Wye-Wye (Yy), Delta-Wye (Dy), or Delta-Delta (Dd)), the types of loads (constant impedance (Z), constant current (I), constant power (P)), whether they contain bus shunts (bus sh.), and whether they have less than full three-phase branches (< 3-p. branch).

TABLE II
FEATURES USED IN IEEE TEST CASES

Case	branch sh.	transformer	loads	bus sh.	< 3-p. branch	ref.
IEEE 13	diag	Yy, Dy	ZIP	yes	yes	[26]
IEEE 34	diag	Yy, Dy	ZIP	yes	yes	[26]
IEEE 123	full	Yy, Dd	ZIP	yes	yes	[26]
LVTestCase	none	Dy	P	no	no	[27]

We include two studies: (i) a comparison of power flow results w.r.t. OpenDSS, using the AC polar form of the power flow equations, and (ii) a demonstration of the proposed abstract OPF problem evaluated using a representative set of formulations from the literature. The first requires support for full matrix branch shunts, transformers and ZIP loads, while the second uses simplified versions of the IEEE cases; transformers are converted to branches, and all loads are converted to wye-connected, constant-power loads. Furthermore, to obtain a non-dimensional voltage profile after simplification within $[0.9, 1.1]$, load set-points were reduced by 50%. Because of the severe load reduction that would have been required to ensure feasibility after the simplification, IEEE123 was omitted from the simplified feeders. The results for AC formulations were obtained with Ipopt v3.12.10 as a solver, while SDP and SOC formulations utilized Mosek v9.0.95.

B. Quality and Run-time Analysis

The key metrics from the feasibility study are presented in Table III. We define the relative error δ as the largest relative difference in voltage magnitude across all network buses,

$$\delta = \max_{i,p} \left| \frac{|U|_{ip}^{\text{OpenDSS}} - |U|_{ip}}{|U|_{ip}^{\text{OpenDSS}}} \right|, \qquad (17)$$

where $|U|_{ip}^{\text{OpenDSS}}$ denotes the voltage magnitude at bus i and phase p obtained from OpenDSS. The only exception is in IEEE123, which contains a floating bus (610), where its phase-to-neutral voltages are only unique up to a constant. Therefore, at the floating bus, the phase-to-phase voltages are instead compared. As reported in Table III, we note that the largest relative difference in voltage magnitude is 1.4e-7, indicating a close match in power flows w.r.t. OpenDSS.

TABLE III
COMPARISON OF POWER FLOW RESULTS OF DISTRIBUTION TEST FEEDER
CASES W.R.T. OPENDSS

				LVTestCase	
	IEEE13	IEEE34	IEEE123	t=500	t=1000
δ	5.1E-8	1.4E-7	1.3E-8	3.2E-8	3.3-8
$\min |U|_{ip}$	0.9750	0.9166	0.9858	1.0353	1.0226
$\max |U|_{ip}$	1.0686	1.0500	1.0437	1.0499	1.0496

Formulation comparison results are presented in Table IV. For each test case we also specify the number of nodes $|N|$ and the number of buses $|E|$. AC-NLP-polar and AC-NLP-rect are equivalent formulations, therefore yielding the same result. The SDP BFM and SOC BFM formulations are relaxations of the nonlinear BFM, equivalent to AC-NLP-polar; so their optimality gap w.r.t. AC-NLP-polar should be positive. For IEEE13, SDP BFM is tight up to the accuracy of the solvers, explaining the negative value. Since SOC BFM is a relaxation of SDP BFM, its optimality gap w.r.t. AC-NLP-polar should be the same or larger; for IEEE13, SDP BFM is tight whilst SOC BFM has a significant gap of 1.1%. It should be noted that AC-NLP-rect and SOC BFM

experienced numerical issues for IEEE13 and LVTestCase, respectively, illustrating the usefulness of the availability of multiple formulations.

VI. CONCLUSIONS

This paper discusses the design and implementation of an open-source toolkit for simulation and optimization of distribution network power flows. This toolbox complements established tools such as OpenDSS and Gridlab-D by providing an optimization-first approach to distribution system modeling. Numerical results presented illustrate that the proposed approach provides comparable numerical results on a variety of unbalanced IEEE test feeders. We further demonstrate how modelers can use PowerModelsDistribution to explore exact models, relaxations and approximations in different variable spaces, and can interface with a wide array of state-of-the-art solvers through the Julia-based mathematical programming package, JuMP. PowerModelsDistribution presents a practical and well-tested domain-specific mathematical programming framework for distribution network optimization. We hope that this package can serve as a foundational tool enabling the speedy development of optimization models for distribution network problems, and, to that end, we encourage the research community to develop and contribute novel problem specifications and mathematical formulations. Furthermore, the extensible design of PowerModelsDistribution lends itself well to becoming a backend optimization tool such as PowerModels has been in *e.g.*, PandaPower [6] and PowerSystems [7].

The development of PowerModelsDistribution is an ongoing effort and in the near future we have plans to support additional high-priority features such as,

- explicit representation of the neutral and/or ground (4-wire), and Kron-reduced representations [28], [16];
- voltage regulators (*i.e.*, auto-transformers) [29], [30];
- short-circuit calculation;
- harmonics analysis [31];
- problem-specific solvers, *e.g.*, power flow algorithms based on Newton-Raphson.

In addition, we hope to explore some experimental features, such as probing the feasibility of solving optimization problems in SI units rather than in the non-dimensional (per-unit) representation we currently utilize, and testing scalability for larger networks, *e.g.*, urban distribution networks.

Finally, this work highlights a need for a library of unbalanced OPF benchmark cases that would enable easy comparison between the various distribution network solvers and simulators, such as the PGlib [32] AC-OPF benchmarks, which are only applicable to single-phase power flow equations.

VII. ACKNOWLEDGEMENTS

REFERENCES

[1] C. Coffrin, R. Bent, K. Sundar, Y. Ng, and M. Lubin, "PowerModels.jl: an open-source framework for exploring power flow formulations," in *Power Syst. Comp. Conf.*, vol. 20, Dublin, Ireland, 2018, p. 8.

[2] I. Dunning, J. Huchette, and M. Lubin, "JuMP: a modeling language for mathematical optimization," *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.

[3] H. Ergun, J. Dave, D. Van Hertem, and F. Geth, "Optimal power flow for AC/DC Grids: formulation, convex relaxation, linear approximation and implementation," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 2980 – 2990, 2019.

[4] C. Coffrin, R. Bent, B. Tasseff, K. Sundar, and S. Backhaus, "Relaxations of AC maximal load delivery for severe contingency analysis," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1450–1458, 2019.

[5] D. K. Molzahn and I. A. Hiskens, "A survey of relaxations and approximations of the power flow equations," *Foundations Trends Electr. Energy Syst.*, 2017.

[6] D. K. Molzahn, B. C. Lesieutre, and C. L. DeMarco, "Approximate representation of ZIP loads in a semidefinite relaxation of the OPF problem," *IEEE Trans. Power Syst.*, vol. 29, no. 4, pp. 1864–1865, 2014.

[7] Z. Shen, Z. Wei, G. Sun, and S. Chen, "Representing ZIP loads in convex relaxations of optimal power flow problems," *Int. J. Elec. Power Energy Syst.*, vol. 110, no. September 2018, pp. 372–385, 2019. [Online]. Available: https://doi.org/10.1016/j.ijepes.2019.03.011

[8] C. Zhao, E. Dall'Anese, and S. Low, "Convex relaxation of OPF in multiphase radial networks with wye and delta connections," in *IREP*, Espinho, Portugal, 2017, pp. 1–7.

[9] B. A. Mather, L. R. de Araujo, S. Paudyal, T. E. McDermott, J. C. Fuller, H. Zhu, W. Kersting, B. C. Pal, G. J. Shirek, L. F. Ochoa, J. L. R. Pereira, R. C. Dugan, C.-W. Ten, K. P. Schneider, and S. Matthias, "Analytic considerations and design basis for the IEEE distribution test feeders," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3181–3188, 2017.

[10] S. Meinecke, M. Braun, F. Meier, A. Scheidler, F. Schafer, J. Dollichon, L. Thurner, and J.-H. Menke, "Pandapower - An open-source python tool for convenient modeling, analysis, and optimization of electric power systems," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, 2018.

[11] R. Berg, E. Hawkins, and W. Pleines, "Mechanized calculation of unbalanced load flow on radial distribution circuits," *IEEE Trans. Power Appar. Syst.*, vol. PAS-86, no. 4, pp. 415–421, 1967.

[12] R. C. Dugan and T. E. McDermott, "An open source platform for collaborating on smart grid research," *IEEE Power Energy Soc. General Meeting*, pp. 1–7, 2011.

[13] D. P. Chassin, K. Schneider, and C. Gerkensmeyer, "Gridlab-d: An open-source power systems modeling and simulation environment," in *2008 IEEE/PES Transmission and Distribution Conference and Exposition*, April 2008, pp. 1–5.

[14] S. P. Vera, M. Lavoie, E. C. Y. Cucho, Bsanjuan, B. Lüers, and A. Murray, "SanPen/GridCal: 100-stars version 3.5," jul 2019. [Online]. Available: https://zenodo.org/record/3334361

[15] "IEC 61968-1 Application integration at electric utilities - System interfaces for distribution management - Part 1: Interface architecture and general recommendations," 2012.

[16] R. M. Ciric, A. P. Feltrin, and L. F. Ochoa, "Power flow in four-wire distribution networks - general approach," *IEEE Trans. Power Syst.*, vol. 18, no. 4, pp. 1283–1290, 2003.

[17] A. Urquhart and M. Thomson, "Assumptions and approximations typically applied in modelling LV networks with high penetrations of low carbon technologies," in *Solar Integration Workshop*. London: Energynautics, 2013, p. 6.

[18] K. Girigoudar, D. K. Molzahn, and L. A. Roald, "Analytical and empirical comparisons of voltage unbalance definitions," in *North American Power Symp.*, Wichita, Kansas, 2019, pp. 1–6.

[19] R. Dugan, "A perspective on transformer modeling for distribution system analysis," in *IEEE Power Energy Soc. General Meeting*, Toronto, Ont., CA, 2003, pp. 114–119.

[20] B. Mahdad, T. Bouktir, and K. Srairi, "A three-phase power flow modelization: a tool for optimal location and control of FACTS devices in unbalanced power systems," in *IEEE IECON*, 2006, pp. 2238–2243.

TABLE IV

OPTIMALITY AND RUNTIME RESULTS FOR VARIOUS UNBALANCED OPF FORMULATIONS. LVTESTCASE EVALUATED AT $t = 1000$.

Test case	$\lvert N \rvert$	$\lvert E \rvert$	\$/h AC-NLP-polar	\$/h AC-NLP-rect	gap (%) SDP BFM	gap (%) SOC BFM	(s) AC-NLP-polar	(s) AC-NLP-rect	(s) SDP BFM	(s) SOC BFM
IEEE 13	38	13	1.7598	n.s.	-2.6E-3	1.1	0.0544	n.s.	0.2196	0.0819
IEEE 34	135	34	0.9557	0.9557	0.47	1.8	0.9106	58.3162	0.6437	0.3317
LVTestCase	2718	906	0.0243	0.0243	0.071	n.s.	4.7184	34.3739	10.8802	n.s.

[21] L. Gan and S. H. Low, "Convex relaxations and linear approximation for optimal power flow in multiphase radial networks," in *Power Syst. Comp. Conf.*, Wroclaw, Poland, 2014, pp. 1–9.

[22] S. Kim, M. Kojima, and M. Yamashita, "Second order cone programming relaxation of a positive semidefinite constraint," *Optimization Methods Software*, vol. 18, no. 5, pp. 535–541, 2003.

[23] M. D. Sankur, R. Dobbe, E. Stewart, D. S. Callaway, and D. B. Arnold, "A linearized power flow model for optimization in unbalanced distribution systems," *[math.OC]*, 2016. [Online]. Available: https://arxiv.org/abs/1606.04492

[24] R. Dobbe, O. Sondermeijer, D. Fridovich-Keil, D. Arnold, D. Callaway, and C. Tomlin, "Data-driven decentralized optimal power flow," *[cs.LG]*, pp. 1–10, 2018. [Online]. Available: http://arxiv.org/abs/1806.06790

[25] S. H. Low, "Convex relaxation of optimal power flow - part I: formulations and equivalence," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 1, pp. 15–27, mar 2014.

[26] W. H. Kersting, "Radial distribution test feeders," *IEEE Transactions on Power Systems*, vol. 6, no. 3, pp. 975–985, Aug 1991.

[27] "Ieee pes amps dsas test feeder working group," http://sites.ieee.org/pes-testfeeders/, accessed: 2019-09-26.

[28] M. Usman, A. Cervi, M. Coppo, F. Bignucolo, and R. Turri, "Bus injection relaxation based OPF in multi-phase neutral equipped distribution networks embedding wye- and delta-connected loads and generators," *Int. J. Elec. Power Energy Syst.*, vol. 114, no. Jan, p. 105394, 2020.

[29] M. Bazrafshan, N. Gatsis, and H. Zhu, "Optimal tap selection of step-voltage regulators in multi-phase distribution networks," *Power Syst. Comp. Conf.*, pp. 1–7, 2018.

[30] ——, "Optimal power flow with step-voltage regulators in multi-phase distribution networks," *[math.OC]*, pp. 1–11, 2019. [Online]. Available: http://arxiv.org/abs/1901.04566

[31] Ying-Yi Hong, "Optimal harmonic power flow," *IEEE Trans. Power Del.*, vol. 12, no. 3, pp. 1267–1274, 2002.

[32] S. Babaeinejadsarookolaee, A. Birchfield, R. D. Christie, C. Coffrin, C. DeMarco, R. Diao, M. Ferris, S. Fliscounakis, S. Greene, R. Huang, C. Josz, R. Korab, B. Lesieutre, J. Maeght, D. K. Molzahn, T. J. Overbye, P. Panciatici, B. Park, J. Snodgrass, and R. Zimmerman, "The power grid library for benchmarking AC optimal power flow algorithms," *[math.OC]*, pp. 1–17, 2019. [Online]. Available: http://arxiv.org/abs/1908.02788